

FHZ

Spherical mirrors and lenses in TikZ – English

tikz-mirror-lens package

<https://www.ctan.org/pkg/tikZ-mirror-lens>

Brazil – December 27, 2022 – Version: 1.0.1

Abstract

This is the documentation for the `tikz-mirror-lens` package. This package allows the automatic drawing of the image of objects in spherical mirrors and lenses from the data of the focus, from the position and height of the object, calculating the position and height of the image, and presenting the notable rays.

Contents

1 Quick start, settings and commands	1	3.2 Ready lens settings	7
2 Gaussian spherical mirror model	2	3.3 Ready lens settings – left	7
2.1 Modeling	2	3.4 Constituent commands	7
2.2 Ready-made mirror setups	2	3.5 Examples of each possible lens case	8
2.3 Constituent commands	3	3.5.1 Convergent	8
2.4 Examples of each possible mirror case	3	3.5.2 Divergent	8
2.4.1 Concave	3	3.6 Equivalence between commands for lenses with objects on the right and on the left	9
2.4.2 Convex	4	3.7 Animation	9
2.5 Animation	5	3.7.1 Convergent	9
2.5.1 Concave	5	3.7.2 Divergent	10
2.5.2 Convex	5	4 Other interesting packages	11
3 Gaussian spherical lens model	6	5 History and versions	11
3.1 Modeling	6		
3.1.1 Object on the right	6		
3.1.2 Objeto à esquerda	6		

1 Quick start, settings and commands

The variables used are:

- `f`: mirror or lens focus;
- `p`: position of the object along the x axis;
- `pp`: position of the image along the x axis;
- `o`: height of the object;
- `i`: image height;
- `epsilon`: absolute distance between p and f ;
- `yM`: mirror height;
- `xL`: axis extension x to the left;
- `xR`: axis extension x to the right;

- **(xC,yC)**: Coordinates of the location of the presented data;
- **arrows**: optional argument to change the density of arrows.

The main commands that create the mirror or lens diagrams based on the object's focus f , position p and height o , in addition to other adjustment parameters, are:

- Mirrors
 - `\mirrorSphGauss[setas]{f}{p}{o}{epsilon};`
 - `\mirrorSphGaussCoord[setas]{f}{p}{o}{epsilon};`
 - `\mirrorSphGaussFixed[setas]{f}{p}{o}{epsilon}{yM}{xL}{xR};`
 - `\mirrorSphGaussFixedCoord[setas]{f}{p}{o}{epsilon}{yM}{xL}{xR}{(xC,yC)};`
- Lenses
 - `\lensSphGauss[setas]{f}{p}{o}{epsilon};`
 - `\lensSphGaussCoord[setas]{f}{p}{o}{epsilon};`
 - `\lensSphGaussFixed[setas]{f}{p}{o}{epsilon}{yM}{xL}{xR};`
 - `\lensSphGaussFixedCoord[setas]{f}{p}{o}{epsilon}{yM}{xL}{xR}{(xC,yC)};`
- Lenses with object on the left
 - For each lens in the previous block, change “Gauss” to “GaussL”.

2 Gaussian spherical mirror model

2.1 Modeling

The equations for the position p' and the height i of the image from the focus f of the mirror and the position p and height o of the object are:

$$\frac{1}{f} = \frac{1}{p} + \frac{1}{p'} \Rightarrow p' = \frac{fp}{p-f}, \quad p \neq f, \quad (1)$$

$$i = -\frac{p'}{p}o.$$

Mirror type definitions are made based on the focus' signal:

$$\begin{aligned} f > 0 &: \text{concave,} \\ f < 0 &: \text{convex.} \end{aligned} \quad (2)$$

Figure 1 presents the definition of the mirror coordinate system, in which $p > 0$ is the position of the object along the x axis and $p' < 0$ is the image position along the x axis. The vertex V of the mirror is the origin of the coordinate system.

2.2 Ready-made mirror setups

Table 1 presents all ready-made mirror configurations provided by the package. The notation is:

- **arrow**: distance between drawn arrows, in case of omission, the default is 60 (pt).
- **epsilon**: distance between object and focus at which the image is not calculated or drawn because it is too big and/or too far from the vertex;
- **yM**: mirror height, either data or calculation;
- **xL**: negative limit of the x axis;
- **xR**: positive limit of the x axis;
- **Co**: the ordered pair (x_C, y_C) of the block of equations that show the focus and coordinates of the object and the image.

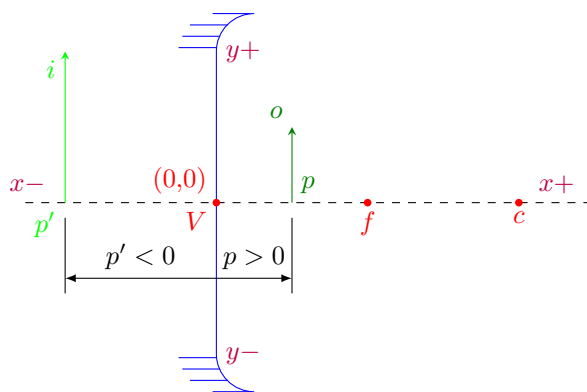


Figure 1: Sign convention for spherical mirrors.

Table 1: All mirror settings ready.

	No	coord	Yes
	<p>$\backslash\text{mirrorSphGauss}[\text{seta}]\{f\}\{p\}\{o\}\{\epsilon\}$</p>	<p>$f = -2.0$ $(p, o) = (3.5, 2)$ $(p', i) = (-1.27277, 0.7273)$</p> <p>$\backslash\text{mirrorSphGaussCoord}[\text{seta}]\{f\}\{p\}\{o\}\{\epsilon\}$</p>	
fixed mirror size	<p>$\backslash\text{mirrorSphGaussFixed}[\text{seta}]\{f\}\{p\}\{o\}\{\epsilon\}\{yM\}\{xL\}\{xR\}$</p>	<p>$f = 2$ $(p, o) = (3.5, 2)$ $(p', i) = (4.9667, -2.66669)$</p> <p>$\backslash\text{mirrorSphGaussFixedCoord}[\text{seta}]\{f\}\{p\}\{o\}\{\epsilon\}\{yM\}\{xL\}\{xR\}\{Co\}$</p>	

2.3 Constituent commands

The command that calculates the position p' and the height i of the image is:

- `\mirrorMath{f}{p}{o}{epsilon}{yM}`.

The following commands draw the main components of the diagram,

- mirror drawing: `\mirrorBase{f}{yM}{xL}{xR};`
- notable points drawing: `\mirrorPts{v}{f}{c};`
- notable rays drawing: `\mirrorRays[arrows]{p}{pp}{o}{i}.`

The following commands are the same for mirrors and lenses, and are responsible for,

- object and image drawing item: `\mirrorLensObjIma{p}{pp}{o}{i}`;
- description of numerical coordinate values: `\mirrorLensCoord{p}{pp}{o}{i}{f}{Co}`.

2.4 Examples of each possible mirror case

2.4.1 Concave

Figures from 2 to 6 present the 5 possible cases of positioning an object in front of a concave mirror.

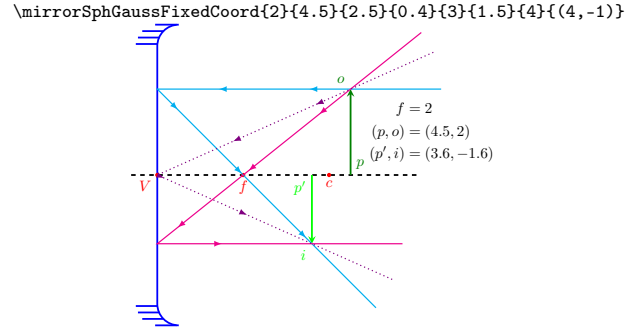


Figure 2: Case 1, object far from the mirror, beyond the center of curvature.

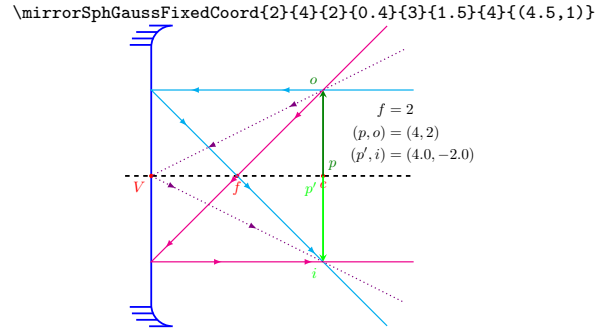


Figure 3: Case 2, object located on the center of curvature.

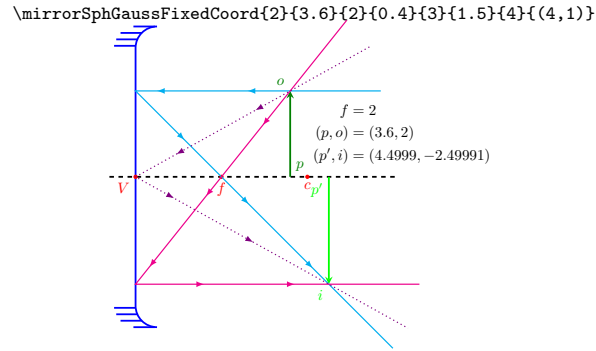
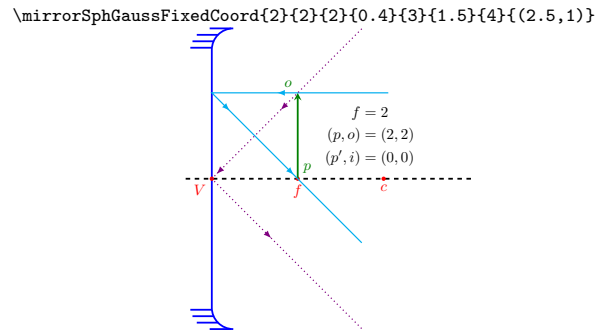


Figure 4: Case 3, object located between the center of curvature and the focus of the mirror.

Figure 5: Case 4, object located on the focus of the mirror (or less than a distance $\varepsilon \rightarrow 0$).

2.4.2 Convex

Figure 7 presents two different positions of the single case of positioning an object in front of a convex mirror.

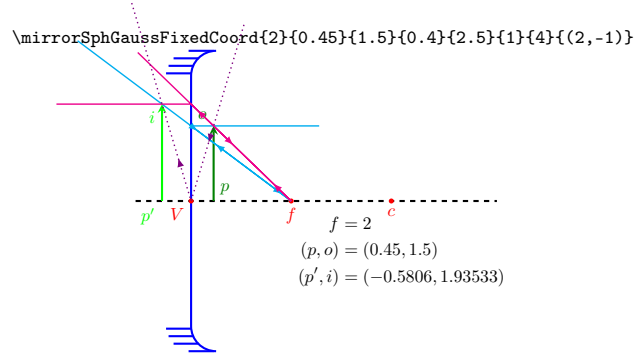


Figure 6: Case 5, object located between focus and mirror vertex.

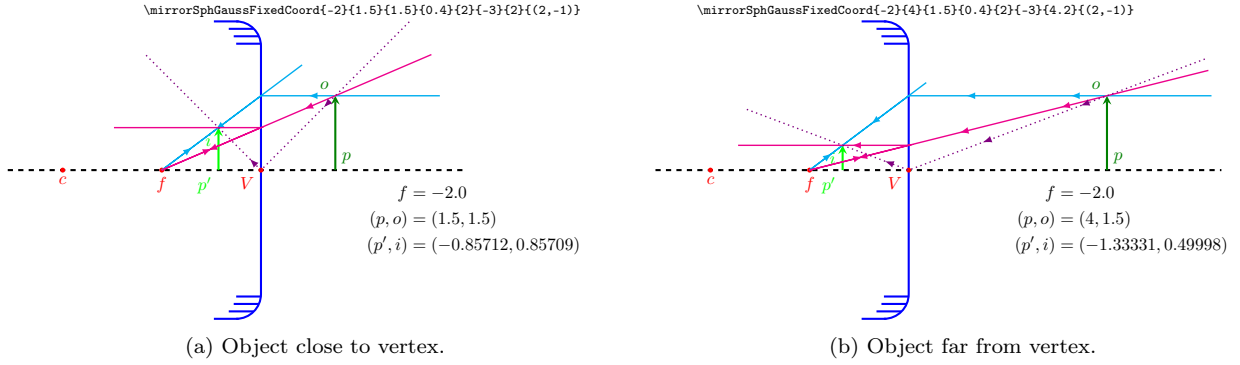


Figure 7: Single case, object located in front of the mirror, at any distance from it.

2.5 Animation

2.5.1 Concave

Figure 8 presents an animation containing the movement of an object close to a concave mirror.

Figure 8: Animation of object approaching a concave mirror.

2.5.2 Convex

Figure 9 presents an animation containing the movement of an object close to a convex mirror.

Figure 9: Animation of object approaching a convex mirror.

3 Gaussian spherical lens model

3.1 Modeling

Figure 10 presents the definition of the lens coordinate system in two cases, the one with the object on the positive side in Figure 10a and with the object on the negative side Figure 10b.

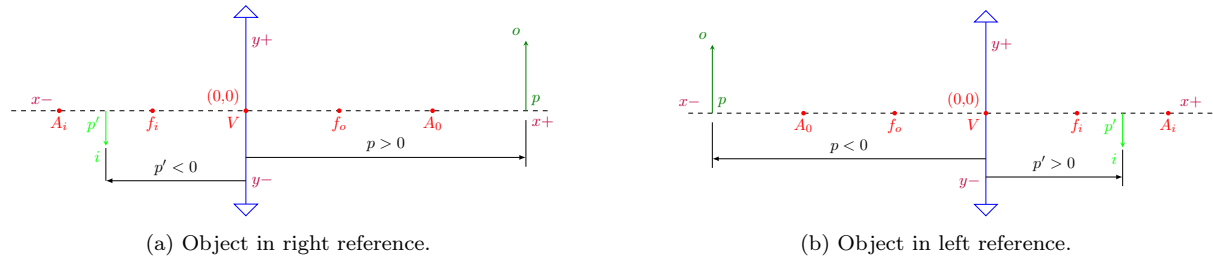


Figure 10: Sign convention for spherical lenses.

Lens type definitions are made based on the focus signal:

$$\begin{aligned} f > 0 : & \text{ convergent,} \\ f < 0 : & \text{ divergent.} \end{aligned} \quad (3)$$

3.1.1 Object on the right

For the object on the right, the easiest way to correct the model from a spherical mirror to a spherical lens is to change the sign of p' .

The equations for the position p' and the height i of the image from the focus f of the mirror, and the position p and height o of the object are:

$$\begin{aligned} \frac{1}{f} &= \frac{1}{p} - \frac{1}{p'} \Rightarrow p' = \frac{fp}{f-p}, \quad p \neq f, \\ i &= \frac{p'}{p} o. \end{aligned} \quad (4)$$

3.1.2 Objeto à esquerda

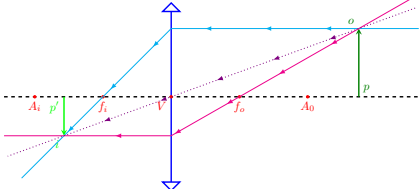
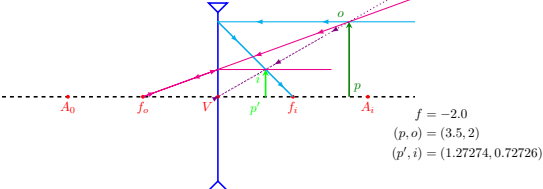
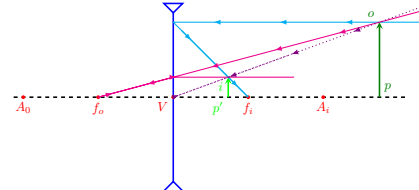
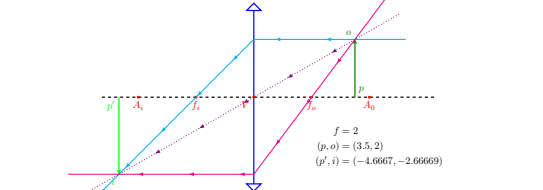
Para o objeto à esquerda, a expressão de p' e i são dadas por:

$$\begin{aligned} \frac{1}{p'} &= \frac{1}{p} + \frac{1}{f} \Rightarrow p' = \frac{fp}{f+p}, \quad p \neq -f, \\ i &= \frac{p'}{p} o. \end{aligned} \quad (5)$$

3.2 Ready lens settings

The Table 2 presents all the ready-made lens configurations provided by the package.

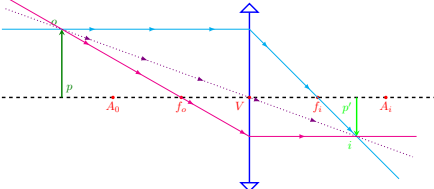
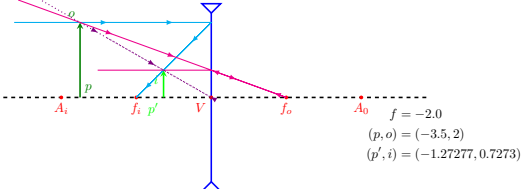
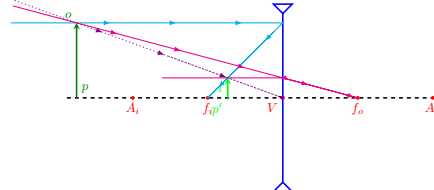
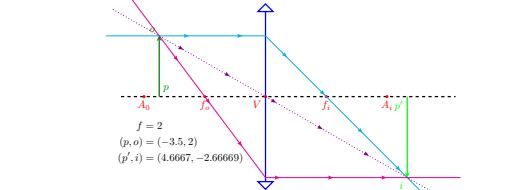
Table 2: All lens settings ready.

		coord	
		No	Yes
fixed mirror size	No	 $\backslash\lensSphGauss[seta]{f}{p}{o}{epsilon}$	 $\backslash\lensSphGaussCoord[seta]{f}{p}{o}{epsilon}$ $f = -2.0$ $(p, o) = (3.5, 2)$ $(p', i) = (1.27274, 0.72726)$
	Yes	 $\backslash\lensSphGaussFixed[seta]{f}{p}{o}{epsilon}{yM}{xL}{xR}$	 $\backslash\lensSphGaussFixedCoord[seta]{f}{p}{o}{epsilon}{yM}{xL}{xR}{Co}$ $f = 2$ $(p, o) = (3.5, 2)$ $(p', i) = (-4.6667, -2.66669)$

3.3 Ready lens settings – left

The Table 3 presents all the ready-made lens settings provided by the package.

Table 3: All lens settings ready with object on left.

		coord	
		No	Yes
fixed mirror size	No	 $\backslash\lensSphGaussL[seta]{f}{p}{o}{epsilon}$	 $\backslash\lensSphGaussLCoord[seta]{f}{p}{o}{epsilon}$ $f = -2.0$ $(p, o) = (-3.5, 2)$ $(p', i) = (-1.27277, 0.7273)$
	Yes	 $\backslash\lensSphGaussLFixed[seta]{f}{p}{o}{epsilon}{yM}{xL}{xR}$	 $\backslash\lensSphGaussLFixedCoord[seta]{f}{p}{o}{epsilon}{yM}{xL}{xR}{Co}$ $f = 2$ $(p, o) = (-3.5, 2)$ $(p', i) = (4.6667, -2.66669)$

3.4 Constituent commands

The command that calculates the position p' and height i of the image with object on the right is:

- $\backslash\lensMath{f}{p}{o}{epsilon}{yM}$.

In turn, the command that calculates the coordinates of the image with the object on the left is:

- `\lensMathL{f}{p}{o}{epsilon}{yM}`,

nonetheless, the change in the nomenclature of the commands that draw the lenses is just the addition of the letter *L* after the word “Gauss”.

The following commands draw the main components of the diagram,

- lens design: `\lensBase{f}{yM}{xL}{xR}`;
- notable points drawing: `\lensPts{v}{f}{a}`;
- notable ray drawing: `\lensRays[arrows]{p}{pp}{o}{i}`.

3.5 Examples of each possible lens case

3.5.1 Convergent

Figures from 11 to 15 present the 5 possible cases of positioning an object in front of a converging lens.

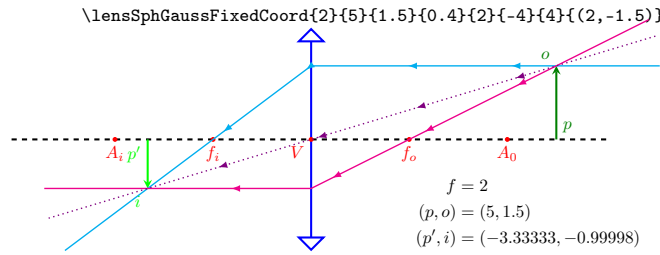


Figure 11: Case 1, object far from mirror, beyond center of curvature.

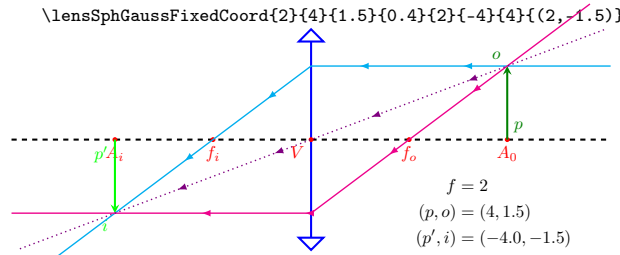


Figure 12: Case 2, object over anti-main object.

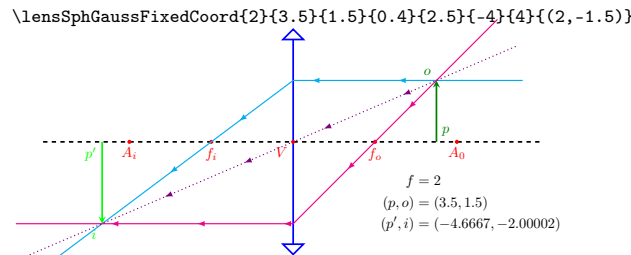


Figure 13: Case 3, object between anti-main object and focus object.

3.5.2 Divergent

Figure 16 presents two different positions of the single case of positioning an object in front of a diverging lens.

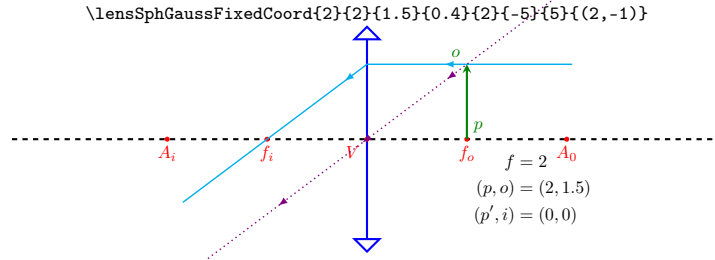
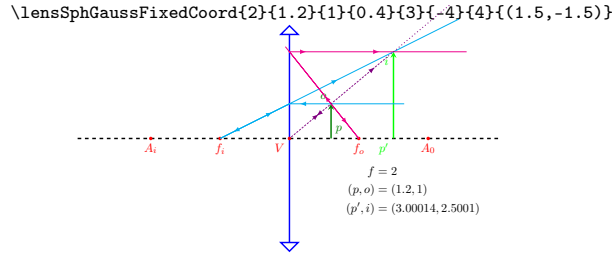
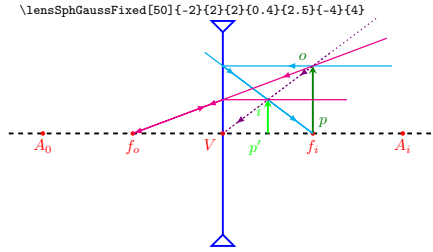
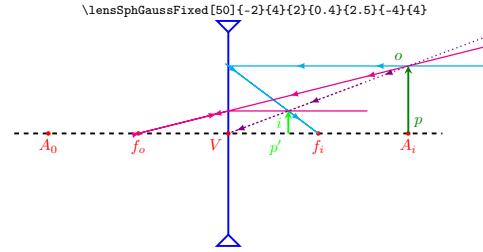
Figure 14: Case 4, object over object focus (or less than a distance $\varepsilon \rightarrow 0$).

Figure 15: Case 5, object between focus object and optical center of lens.



(a) Between focus and vertex.

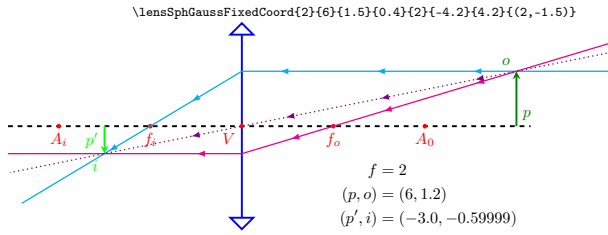


(b) Beyond the center of curvature.

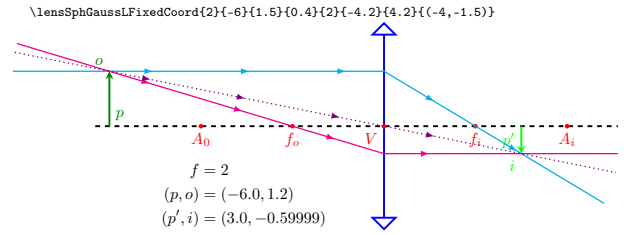
Figure 16: Single case, object located in front of the lens, at any distance from it.

3.6 Equivalence between commands for lenses with objects on the right and on the left

Figure 17 presents the equivalence between the commands that calculate and draw the image using converging lenses depending on the location of the object.



(a) Command for object on the right.



(b) Command for object on the left.

Figure 17: Equivalence between commands for converging lenses.

Figure 18 presents the equivalence between the commands that calculate and draw the image using diverging lenses depending on the location of the object.

3.7 Animation

3.7.1 Convergent

Figure 19 presents an animation containing the movement of an object close to a converging lens.

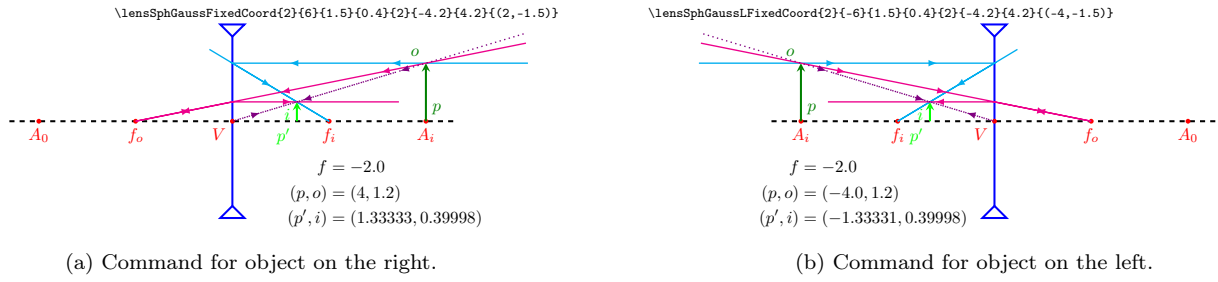


Figure 18: Equivalence between commands for diverging lenses.

Figure 19: Animation of object approaching a converging lens.

3.7.2 Divergent

Figure 20 presents an animation containing the movement of an object close to a diverging lens.

Figure 20: Animation of object approaching a diverging lens.

4 Other interesting packages

Below are interesting *links* to other packages with optics implementations, as well as sources for the equations and modeling used.

1. [TeX StackExchange](#) – TikZ library for optics?
2. [TeX StackExchange](#) – Geometrical optics
3. [CTAN](#) – tikz-optics
4. [CTAN](#) – pst-mirror
5. [CTAN](#) – simpleoptics
6. [YouTube](#) – The Organic Chemistry Tutor – Spherical Mirrors & The Mirror Equation - Geometric Optics
7. [hyperphysics](#) – Spherical Mirror Equation
8. [hyperphysics](#) – lenseq
9. [plymouth](#) – lenses
10. [khanacademy](#) – lens formula

5 History and versions

1.0.0 (2022-12-24): Package creation.

1.0.1 (2022-12-27): Small corrections on function argument input order in `\mirrorRays` and in `\lensRays`.